>cm focus

CHIPTUNES

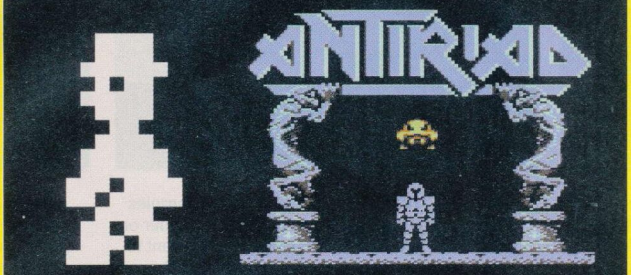
The lo-fi bleeps of vintage videogames have inspired a whole musical scene, but they're also used in modern pop. Here's how...

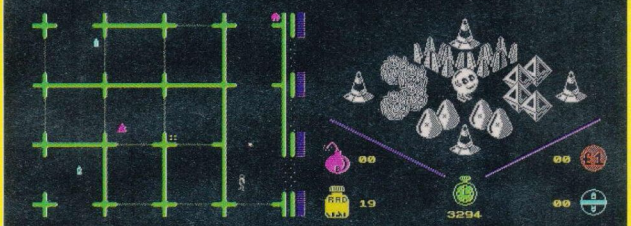
cm DVD

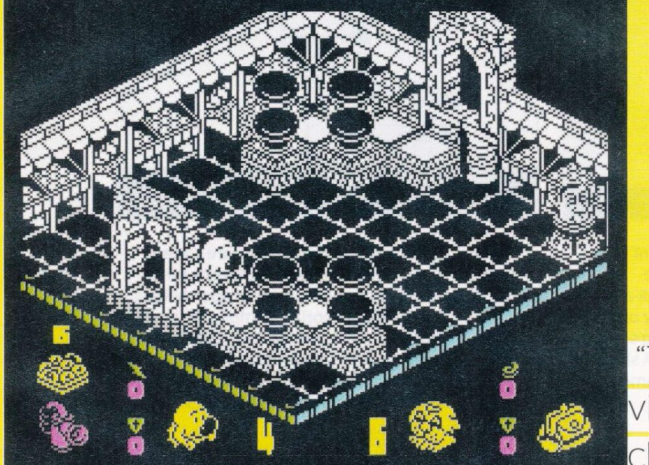
ON THE DVD

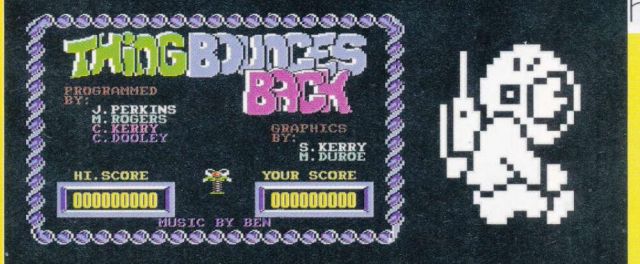
TUTORIAL FILES

Our Cubase 5 chiptune project is in the **Tutorial Files** folder; Unkown 64 CM is in the **CM Studio**









> For those who grew up in the 80s and 90s, nothing reeks of nostalgia like the lo-fi sounds of early videogames. 8-bit machines like the Sinclair ZX Spectrum, Commodore 64 and Nintendo NES might have had very limited sound capabilities, but the computer musicians of the day managed to squeeze some classic tunes out of them.

The spirit of these old-skool soundtracks lives on in the thriving retro-centric chiptune scene of today. But what defines chiptune? The term itself originated in the early 90s when the Amiga demoscene's musicians would create 80s-style video game music using trackers like SoundTracker and ProTracker. Unlike the systems they emulated, the Amiga used samples instead of oscillators-on-a-chip (like the C64, Spectrum, etc), so to imitate this, compact 'chiptunes' used crude samples, sometimes cycling a single waveform to create sounds, and were thus small enough to include in size-critical demos and other pieces of software where space was at a premium.

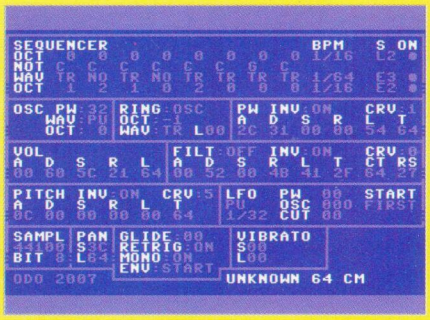
Eventually, the subtly different term 'chip music' came to be used to describe 'the real thing' – ie, music actually made on, say, a ZX Spectrum or C64, using the sound chip for synthesis/sound duties.

These days, however, the term has become far less specific, and anything that exhibits a whiff of vintage videogame music is labelled 'chiptune', and the influence of early videogame soundtracks and the chiptune scene can be clearly heard in mainstream pop music.

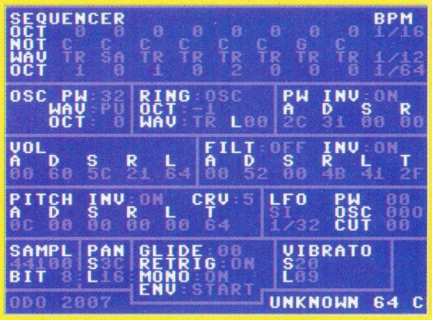
> "The influence of videogames and the chiptune scene can be heard in pop music"

In this **cm** *Focus*, we're going to look at how you can get an authentic chiptune-style sound using the tools available in the modern DAW. By employing techniques like fast arpeggiation and sample rate reduction, we can resurrect the bleeps and bloops of the 80s. By limiting ourselves to basic synth sounds and just a few tracks, we encourage ourselves to compose in a different way, which can lead to musical, sound design and mixing ideas that we might not otherwise have stumbled upon. Now, without further ado, let's get this bleeping show on the road! **cm**

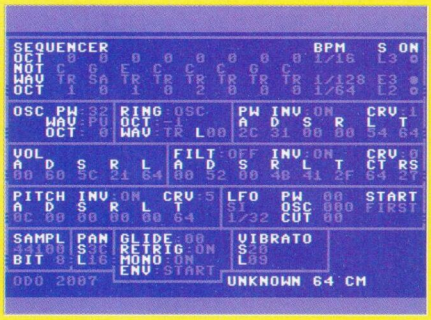
> Step by step Creating arcade-style arps



1 > One crafty technique used by old-skool video game composers was to create the illusion of chords by rapidly cycling a single voice through different notes at very high speed. This psychedelic effect instantly brings mid-80s videogames to mind. Load Unknown 64 CM (on the DVD) onto a new track – a good starting point for a lead sound is the **Lead:Screemer** patch.



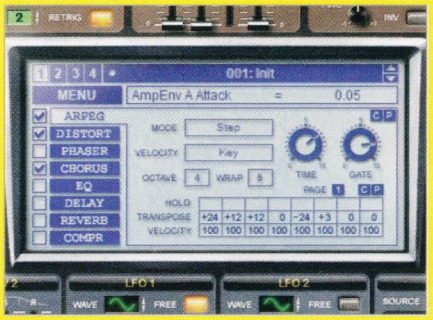
2 > To turn up the speed of the **Wave** sequencer, set its **BPM** to **1/128** and change the second step to **SA**w. Adjust the play mode to **E3**, and set the bottom **Oct**ave sequencer's first two slots to **1** and **0**, and the play mode to **L2** so it loops round them. Set its **BPM** to **1/64**. This will give you a very fast lead sound.



3 > Save this patch and load another Unknown 64 CM. Load up the saved patch and change the first row of the arpeggiator (**Oct**) to **C**, **G**, **E** with a loop setting of **L3**. This gives us a version of the same arp that plays a major chord rather than just alternating octaves.

> Step by step Making simple synth leads

1 > Memorable bleepy synth leads are a staple of the chiptune aesthetic. To get a really simple lo-fi synth sound, you can use pretty much any virtual analogue synth, using one of the oscillators set to a single-voice waveform. In this case, we're using Sylenth1 for a sawtooth sound. Adding a mono-to-stereo effect widens the sound, which can be useful for fitting it in the mix.



2 > Now we'll create a big arpeggio for the intro. Make a new instance of Sylenth1 and copy the settings we've used here. This patch employs a two-voice detuned sawtooth with a two-voice detuned square wave an octave above it, and a 16th-note arpeggiator that plays over four octaves. Copy the step sequencer settings we've used here.

3 > To create a variation on this arp, save out the patch and bring up another **Sylenth1**. Load up the patch and change the first oscillator's waveshape from sawtooth to square. Also, turn the **Gate** knob in the Arpeggiator section down to **44.76%**. This gives the arp a shorter, more aggressive, stabby feel.

4 > Our final lead sound is based on a pair of two-voice square wave oscillators, with one of them detuned slightly. Run this through a **1/64th** note arp effect set to **4 Octaves** and a **50% Gate** time. Change the **Distort**ion to **Decimate Mode** with a **2.19 Amount** setting.

Visit the chip shop

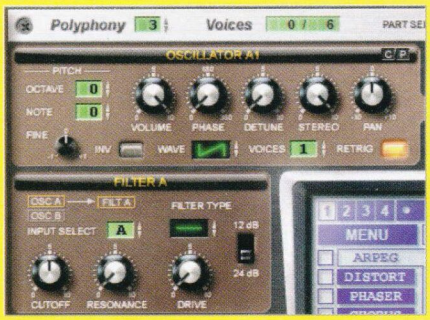
There are tons of places on the web that you can go to check out chiptune music. For starters, we recommend the High Voltage SID Collection (hvsc.c64.org), containing a mass of Commodore 64 music, including classic game soundtracks and more recent compositions. You'll need a SID file player, such as sidplay2/w for Windows (www.gsldata.se/c64/spw) or SIDPlay for Mac (www.sidmusic.org/sidplay/mac). If you want a slightly more convenient way to check out some SID music, you can find MP3 versions of SID tracks recorded using a HardSID card (www.hardsid.com) at The SID 6581/8580 Recordings Archive (sid.oth4.com).

For Amiga chiptunes, you can find an enormous collection at amp.dascene.net, with yet more mods at modarchive.org, micromusic.net and www.chiptune.com. The easiest way to hear the files is to use a module player, such as XMplay.

If you aren't sure quite where to start, try the kohina internet radio station (www.kohina.com). This site features a convenient in-browser player, alongside streams of various formats, and a huge catalogue of chiptunes to get your lugholes around.

Other oddities that you might find interesting include kindofbloop.com, which features shocking chiptune versions of tracks from Miles Davis' classic jazz album *Kind of Blue* and some Daft Punk covers, located at thelemurblog.com/2009/07/15/da-chip.

> Step by step Lo-fi 'vocal' bass



1 > Sample rate reduction effects can be great for making vocal-sounding bass noises. Load up a virtual analogue synth – we're using Sylenth1 – and set one oscillator to a sawtooth shape, and another oscillator down an **Octave** to a square shape. This gives us a nice, chunky bass tone to play with.

2 > Set filter A to low-pass, and turn the **Cutoff** down and **Resonance** and **Drive** up, as shown. Route the first envelope to the filter **Cutoff AB**, and dial in a relatively slow **A**ttack, so that the filter opens up over the course of the sound.

3 > Adding a chorus effect can warm the sound up, but the most important effect is the sample rate reduction, which gives the sound its vocal-like quality. Here we set the Cubase BitCrusher's **SampleDivider** knob to **22**, which gives the sound a really nasty lo-fi feel – in fact, this instantly gives almost any sound added chiptune authenticity!

Further reading and viewing

8-BIT SOUNDS IN REASON TUTORIAL
bit.ly/13Ui3i
This guide to making chiptune sounds in Propellerhead Reason show you how to create your own percussion and other 8-bit-style sounds with Subtractor.

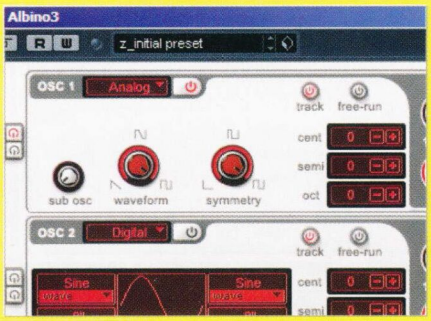
RADIOHEAD REMIX - *CREEP (8BIT VERSION)*
bit.ly/XUtAN
Radiohead's early 90s classic gets re-imagined in a chiptune style. Self-loathing never sounded so bouncy!

MARTIN GALWAY - *PARALLAX*
bit.ly/2fyHXH
This C64 game from *Sensible Soccer* and *Cannon Fodder* creators Sensible Software is a classic, and Martin Galway's epic, prog-tastic theme still sounds great to this day.

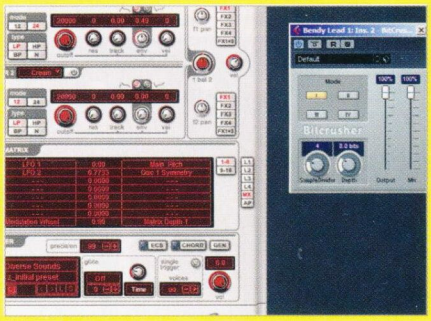
MARIO PAINT COMPOSER
www.unfungames.com/mariopaint/
This PC and Mac recreation of the music composition component of Nintendo's Mario Paint is ideal for those looking to create authentic SNES-style music in a stave-based environment.

THE HARDWARE CHIPTUNE PROJECT
bit.ly/kbIpe
This video shows a slammin' chiptune running in its entirety from a single microcontroller chip

> Step by step Pulse-width modulation leads



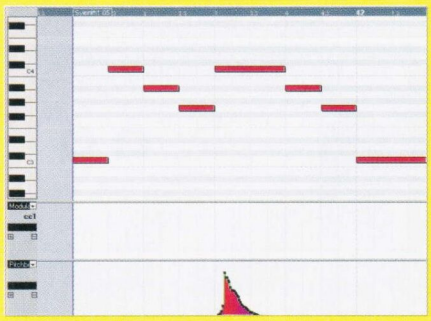
1 > Old-skool audio chips had only a few voices to play with, so creating big sounds using multiple oscillators was rarely an option. An easy way to get extra harmonics from a single oscillator is to use pulse width modulation, á la the C64's SID chip. Here we're using Albino 3. Load the **Z_init** preset from the **Diverse Sounds** category and set the first oscillator's **Waveform** to pulse.



2 > Next, bring up LFO2's control panel and turn off the tempo sync. Set the LFO **Rate** to **3.56**. Return to the modulation matrix and set the second line to read **LFO 2 0.7733 Osc 1 Symmetry**. This modulated the symmetry (or pulse width in this case) of oscillator one with LFO 2. Now throw on a BitCrusher – we've gone for the built-in Cubase plug-in with the **SampleDivider** knob at **4**.

MOD MATRIX
LFO 1 0.00 Main Pitch
LFO 2 0.7799 Osc 1 Symmetry
LFO 2 0.8066 Osc 2 Symmetry
Modulation Wheel 0.90 Matrix Depth 1
MASTER
Diverse Sounds

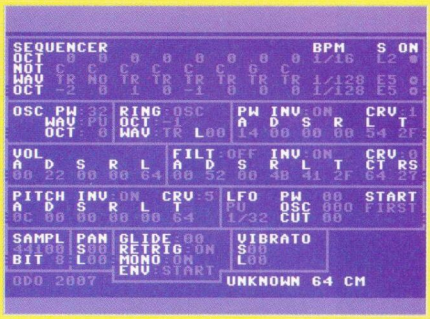
3 > To create an alternative version of the lead, save the patch and load it up into another Albino 3 – activate oscillator 2 and set it to a pulse **Waveform**. Tune this oscillator up seven **semi**tones to create a perfect fifth, then set the third line of the modulation matrix to make **LFO 2** affect **Osc 2 Symmetry** by **0.8066**.



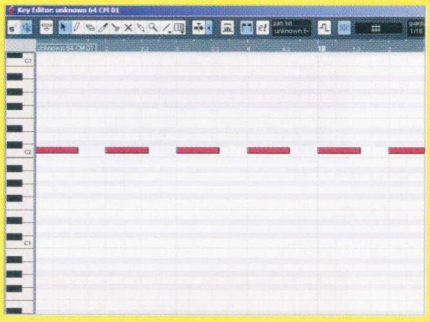
4 > When programming the MIDI parts for these lead synth sounds, you can use pitchbend to get a more expressive feel. Also, by default, Albino routes the modulation wheel to vibrato amount, and programming in mod wheel curves creates wild 80s-style vibrato effects.

> Step by step Drums and percussion

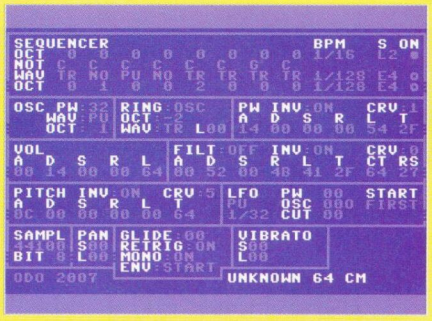
1 > Not only are arpeggiators useful for creating monophonic 'chords', they can help us get a variety of other sounds, too – even percussion. Begin by loading up Unknown 64 CM in your DAW, and selecting the preset **PER:Whaat**. Play a note – it's a thumping kick drum.



2 > Check out the arpeggiator section (called Sequencer). Two of the three arpeggiator components are **On**: the waveshape and secondary operator sections. These are set to **1/128** – very fast indeed! The sequence lasts for five steps, cycling through a **-2 Oct**ave **Tri**angle **Wav**e, white **No**ise, a **+1 Oct**ave **Tri**angle **Wav**e, a root note **Tri**angle **Wa**ve and finally a **-1 Octave Tri**angle **Wav**e.

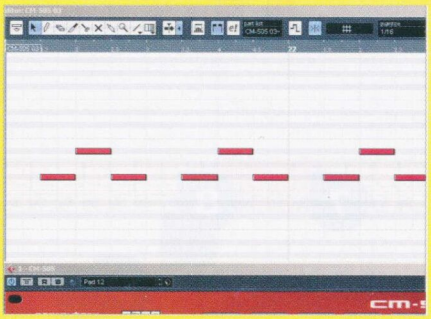


3 > The white noise provides the sound's attack, with the movement from high to low triangle waves providing the typical 'meat' of the kick drum tone. The **-2 Tri**angle **Wa**ve at the start gives the sound a bit of low-end thump. Now program a simple 4/4 kick drum pattern, as shown.



4 > Open up another Unknown 64 CM and this time select the **PER:Known Snare** patch. Note that, while the bottom row of the arpeggiator is active on this patch, it doesn't actually have any affect on the sound, because the only steps that it changes the pitch on are set to **No**ise **Wav**es. To create the snare sound, the synth simply cycles through **Tri**angle, **No**ise, **Pu**lse and **No**ise **Wav**eforms.

5 > Sequence the snare on every second beat, as we have here. To give the sound extra width, add a 'stereoiser' effect, such as Cubase's **MonoToStereo**. This gives the mono synth sound a pleasing stereo spread, and stops it occupying the same space as the kick, which sits firmly at the centre of the mix.



6 > Let's add some more percussion courtesy of **CM-505** (on the DVD). Program this simple beat using the default drum kit. The sound is too hi-fi to pass for chiptune, so add a sample rate reduction effect. Here we've used Cubase's **BitCrusher**, with the **SampleDivider** set to **7** to get the effect of an appropriately low sample rate.

Chiptune in other genres

RUSKO – *BIONIC COMMANDO*
bit.ly/3aVqX
A rinsing dubstep interpretation of the *Bionic Commando* theme, commissioned to promote the 3D remake of Capcom's 80s arcade classic. The track is also available in DnB and electro flavours, remixed by Clipz and Surkin respectively.

BECK – *BAD CARTRIDGE*
bit.ly/hJP5E
Who would have thought mentalist lo-fi funkster Beck would have a soft spot for videogame sounds? His album *Guero* is laced with such audio references, and this remix of *E-Pro* takes the sound to its logical, chip-tastic conclusion.

ZOMBIE NATION – *KERNKRAFT 400*
bit.ly/176HkP
Sampling a refrain from David Whittaker's soundtrack to the 8-bit game *Lazy Jones*, this dirty slice of electro house brought the sound of chiptune to the pop charts and mainstream dance floors.

NELLY FURTADO – *DO IT*
bit.ly/dPbMk
Timbaland's SID chip-sampling, controversy-causing classic might sound uncomfortably similar to *Acidjazzed Evening*, but legal wrangles aside, it's an undeniably catchy slice of pop music.

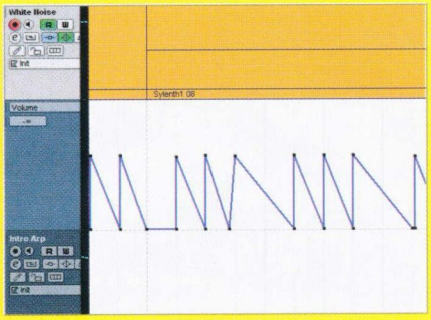
50 CENT FT. JUSTIN TIMBERLAKE & TIMBALAND – *AYO TECHNOLOGY*
bit.ly/RhjmI
More arpeggiator-powered riffs from Timbaland, with an epic sound that's truly retro and futuristic at the same time.

SUPER COCOA BRUVAS – *SUPER BROOKLYN*
bit.ly/Di2OD
The NYC rappers use the sounds of the original *Super Mario Brothers* (specifically the ultra-funky underground cave theme) as the backing to their hardcore rhymes.

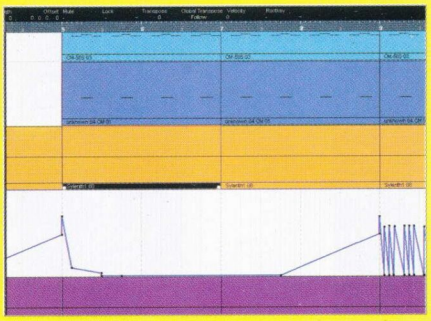
POWER PILL – *PAC MAN*
bit.ly/9SrOH
This cheeky track from Aphex Twin is a relatively classy example of the spate of poppy rave tracks from the early 90s that rehashed classic game music. See Ambassadors of Funk featuring MC Mario's *SuperMarioLand*, Dr Spin's *Tetris* and Hedgehogs With Attitude's *Supersonic* to hear its less tolerable contemporaries.

> Step by step Drums and percussion continued

7 > We need to get some more hats and cymbal sounds in there. Thankfully, lo-fi versions of these noises are easy to create – you just need a synth with a built-in noise generator, such as Lennar Digital's Sylenth1. Set Oscillator A1's **Wave** to a noise shape, and create a MIDI part that plays for the whole bar.

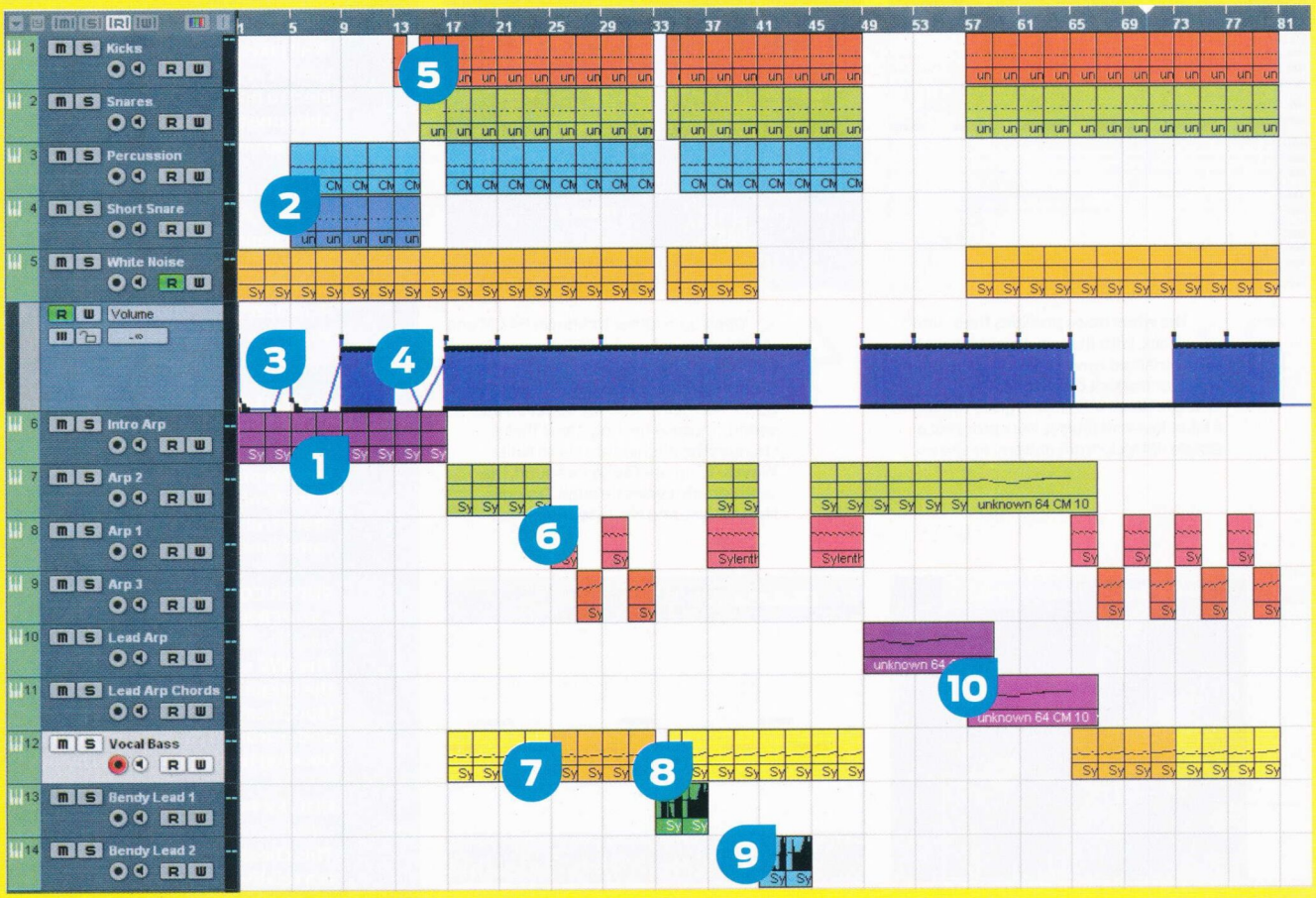


8 > When this is played back, you'll get a wall of noise. To turn this into useful percussion, you just need to automate the track's volume level. Here, we've created open and closed hi-hat effects using downward ramps of various lengths. Add a **BitCrusher** effect to reduce the sample rate and make the sound more lo-fi: we've gone for a **SampleDivider** rate of **2**.



9 > You can also use the white noise to create cymbal crashes and other similar sounds. Here, you can see how the volume level falls quickly, then slowly, to create a typical crash effect – it then slowly rises back up to generate an approximation of a reverse crash.

Putting it all together



1 The intro kicks of with the simple, relatively slow arpeggiator lead, which entices the listener with its lo-fi charm.

2 Before the beat kicks in, a pared-down version of the percussion pattern is introduced, with a less dominant kick sound, so as not to over-power the intro.

3 These slow volume envelopes on the white noise channel create cymbal and reverse crash effects.

4 Yet longer envelopes are used to get a bigger, more sweeping noise effect that builds up to the drop.

5 A simple drum fill leads us nicely into the drop. Once there, the white noise ducks out quickly when the snare plays to give the sound added impact.

6 To create a bit of variety, the track switches between different arp sounds, helping maintain the listener's interest.

7 The bassline changes to complement the different arp patterns at this stage, denoted by the variety of differently coloured MIDI parts.

8 This cheeky gap hints at a breakdown, but the beats and bass kick back in after a single bar's respite.

9 These bendy leads use the chiptune staples of portamento and pitchbend to get their expressive sound. Note that the second version has an added fifth.

10 The second lead rapidly cycles through the notes of a major chord to create a bouncy crescendo.